Given a palindromic string of lowercase English letters palindrome, replace **exactly one** character with any lowercase English letter so that the resulting string is **not** a palindrome and that it is the **lexicographically smallest** one possible.

Return *the resulting string. If there is no way to replace a character to make it not a palindrome, return an* ***empty string****.*

A string a is lexicographically smaller than a string b (of the same length) if in the first position where a and b differ, a has a character strictly smaller than the corresponding character in b. For example, "abcc" is lexicographically smaller than "abcd" because the first position they differ is at the fourth character, and 'c' is smaller than 'd'.

**Example 1:**

Input: palindrome = "abccba"  
Output: "aaccba"  
Explanation: There are many ways to make "abccba" not a palindrome, such as "zbccba", "aaccba", and "abacba".  
Of all the ways, "aaccba" is the lexicographically smallest.

**Example 2:**

Input: palindrome = "a"  
Output: ""  
Explanation: There is no way to replace a single character to make "a" not a palindrome, so return an empty string.

**Constraints:**

* 1 <= palindrome.length <= 1000
* palindrome consists of only lowercase English letters.